

In the Claims

Claims 1-58 remain in the application for consideration and are listed below:

1. **(ORIGINAL)** A computer-implemented architecture comprising:
one or more first objects that support only static properties; and
one or more second objects associated with the one or more first objects
and configured to call the one or more first objects to effect one or more property
value changes on the one or more first objects in a manner that makes the one or
more first objects appear as if they support dynamic properties.

2. **(ORIGINAL)** The computer-implemented architecture of claim 1,
wherein the one or more second objects are configured to maintain property data
that is used to call the one or more first objects.

3. **(ORIGINAL)** The computer-implemented architecture of claim 2,
wherein the property data comprises at least one property value change that is to
be made.

4. **(ORIGINAL)** The computer-implemented architecture of claim 2,
wherein the property data comprises a time at which a property value change is to
be made.

5. **(ORIGINAL)** The computer-implemented architecture of claim 2,
wherein the property data comprises how a property value change is to be made.

1 6. **(ORIGINAL)** The computer-implemented architecture of claim 2,
2 wherein the property data comprises one or more of the following: at least one
3 property value change that is to be made, a time at which a property value change
4 is to be made, and how a property value change is to be made.

5
6 7. **(ORIGINAL)** The computer-implemented architecture of claim 2,
7 wherein the property data comprises at least one property value change that is to
8 be made, a time at which a property value change is to be made, and how a
9 property value change is to be made.

10
11 8. **(ORIGINAL)** The computer-implemented architecture of claim 1
12 further comprising one or more data structures associated with the one or more
13 second objects, individual data structures containing data that is to be used by the
14 one or more second objects to effect a property value change.

15
16 9. **(ORIGINAL)** The computer-implemented architecture of claim 8,
17 wherein the one or more data structures comprise an array of one or more sets of
18 data structures, each set of data structures being associated with a property that is
19 to be changed and containing property data that is to be used to change property
20 values for a property.

21
22 10. **(ORIGINAL)** The computer-implemented architecture of claim 9,
23 wherein the property data comprises at least one property value change that is to
24 be made.

1 **11. (ORIGINAL)** The computer-implemented architecture of claim 9,
2 wherein the property data comprises a time at which a property value change is to
3 be made.

4
5 **12. (ORIGINAL)** The computer-implemented architecture of claim 9,
6 wherein the property data comprises how a property value change is to be made.

7
8 **13. (ORIGINAL)** The computer-implemented architecture of claim 9,
9 wherein the property data comprises at least one property value change that is to
10 be made, a time at which a property value change is to be made, and how a
11 property value change is to be made.

12
13 **14. (ORIGINAL)** Software code embodied on a computer-readable
14 medium which, when executed by a computer, implements the system of claim 1.

15
16 **15. (ORIGINAL)** A multi-media editing application comprising the
17 computer-implemented system of claim 1.

18
19 **16. (ORIGINAL)** A multi-media project editing architecture
20 comprising:

21 one or more first objects that support only static properties, the one or more
22 first objects being configured to implement a transform associated with processing
23 of a multi-media editing project;

24 one or more second objects associated with the one or more first objects
25 and configured to call the one or more first objects to effect one or more property

1 value changes on the one or more first objects in a manner that makes the one or
2 more first objects appear as if they support dynamic properties; and

3 one or more data structures associated with the one or more second objects,
4 individual data structures containing property data that is to be used by the one or
5 more second objects to effect a property value change.

6
7 **17. (ORIGINAL)** The multi-media project editing architecture of claim
8 16, wherein the one or more data structures comprise an array of one or more sets
9 of data structures, each set of data structures being associated with a property
10 whose values are to be changed and containing property data that is to be used to
11 change property values for a property.

12
13 **18. (ORIGINAL)** The multi-media project editing architecture of claim
14 17, wherein the property data comprises at least one value to which a property is to
15 be changed.

16
17 **19. (ORIGINAL)** The multi-media project editing architecture of claim
18 17, wherein the property data comprises a time at which at least one property
19 value is to be changed.

20
21 **20. (ORIGINAL)** The multi-media project editing architecture of claim
22 17, wherein the property data comprises how at least one property value is to be
23 changed.

1 **21. (ORIGINAL)** The multi-media project editing architecture of claim
2 17, wherein the property data comprises: at least one value to which a property is
3 to be changed, a time at which at least one property value is to be changed, and
4 how at least one property value is to be changed.

5
6 **22. (ORIGINAL)** Software code embodied on a computer-readable
7 medium which, when executed by a computer, implements the system of claim 16.

8
9 **23. (ORIGINAL)** A multi-media editing application comprising the
10 computer-implemented system of claim 16.

11
12 **24. (ORIGINAL)** A multi-media project editing architecture
13 comprising:

14 a software-implemented matrix switch having multiple input pins and
15 multiple output pins, the multiple input pins being routable to the multiple output
16 pins, the switch being configured to provide a data stream that represents a multi-
17 media project;

18 a data structure associated with the matrix switch and configured for use in
19 programming the matrix switch to provide a routing scheme for routing input pins
20 to output pins;

21 one or more first objects associated with the matrix switch, the one or more
22 first objects supporting only static properties associated with rendering of a multi-
23 media project;

24 one or more second objects associated with the one or more first objects
25 and configured to call the one or more first objects to effect one or more property

1 value changes on the one or more first objects in a manner that makes the one or
2 more first objects appear as if they support dynamic properties.

3
4 **25. (ORIGINAL)** The multi-media project editing architecture of claim
5 24 further comprising one or more data structures associated with the one or more
6 second objects, individual data structures containing data that is to be used by the
7 one or more second objects to effect a property value change.

8
9 **26. (ORIGINAL)** The multi-media project editing architecture of claim
10 25, wherein the one or more data structures comprise an array of one or more sets
11 of data structures, each set of data structures being associated with a property
12 whose values is to be changed and containing property data that is to be used to
13 change property values.

14
15 **27. (ORIGINAL)** The multi-media project editing architecture of claim
16 26, wherein the property data comprises a property value of a property that is to be
17 changed.

18
19 **28. (ORIGINAL)** The multi-media project editing architecture of claim
20 26, wherein the property data comprises a time at which a property value is to be
21 changed.

22
23 **29. (ORIGINAL)** The multi-media project editing architecture of claim
24 26, wherein the property data comprises how a property value is to be changed.

1 **30. (ORIGINAL)** The multi-media project editing architecture of claim
2 26, wherein the property data comprises a property value of a property that is to be
3 changed, a time at which a property value is to be changed, and how a property
4 value is to be changed.

5
6 **31. (ORIGINAL)** A property value-changing method comprising:
7 providing one or more objects that support only static properties;
8 providing one or more programmable objects configured to effect property
9 value changes on the objects that support only static properties; and
10 effecting at least one property value change on the one or more objects that
11 support only static properties using the one or more programmable objects.

12
13 **32. (ORIGINAL)** The method of claim 31 further comprising
14 programming the one or more programmable objects with property data that is to
15 be used by the one or more programmable objects to effect said at least one
16 property value change.

17
18 **33. (ORIGINAL)** The method of claim 32, wherein the property data
19 comprises one or more property values that are to be changed.

20
21 **34. (ORIGINAL)** The method of claim 32, wherein the property data
22 comprises a time at which a property value is to be changed.

23
24 **35. (ORIGINAL)** The method of claim 32, wherein the property data
25 comprises how a property value is to be changed.

1
2 **36. (ORIGINAL)** The method of claim 32, wherein the property data
3 comprises one or more property values that are to be changed, a time at which a
4 property value is to be changed, and how a property value is to be changed.
5

6 **37. (ORIGINAL)** The method of claim 32 further comprising
7 organizing the property data in one or more data structures that are used by the one
8 or more programmable objects.
9

10 **38. (ORIGINAL)** The method of claim 32 further comprising
11 organizing the property data in one or more data structures that are used by the one
12 or more programmable objects, said organizing comprises defining an array of
13 data structures, each array comprising one or more sets of structures and each set
14 being associated with a property whose value can change.
15

16 **39. (ORIGINAL)** The method of claim 31, wherein said effecting
17 comprises calling the one or more objects that support only static properties with
18 the one or more programmable objects.
19

20 **40. (ORIGINAL)** One or more computer-readable media having
21 computer-readable instructions thereon which, when executed by a computer,
22 implement the method of claim 31.
23

24 **41. (ORIGINAL)** A property value-changing method comprising:
25

1 programming a programmable object with property data that defines when
2 certain property value changes are to be made and what those property value
3 changes are;

4 calling, with the programmable object, one or more objects that do not
5 support dynamic properties; and

6 responsive to said calling, using the property data to effect a property value
7 change on the one or more objects that do not support dynamic properties.

8
9 **42. (ORIGINAL)** The method of claim 41 further comprising calling
10 the programmable object with a time value, the programmable object using the
11 time value to ascertain when to call the one or more objects.

12
13 **43. (ORIGINAL)** The method of claim 41, wherein said programming
14 comprises arranging the property data in a data structure array comprising one or
15 more sets of data structures, each set of data structures being associated with a
16 property whose value is to be changed.

17
18 **44. (ORIGINAL)** One or more computer-readable media having
19 computer-readable instructions thereon which, when executed by a computer,
20 implement the method of claim 41.


21
22 **45. (ORIGINAL)** One or more computer-readable media having
23 computer-readable instructions thereon which, when executed by a computer,
24 cause the computer to:

25 provide one or more objects that support only static properties;

1 provide one or more programmable objects configured to effect property
2 value changes on the objects that support only static properties;

3 program the one or more programmable objects with property data that is to
4 be used by the one or more programmable objects to effect said at least one
5 property value change, the property data comprising: property value changes that
6 are to be made, time(s) at which property value changes are to be made, and how
7 the property value changes are to be made; and

8 effect at least one property value change on the one or more objects that
9 support only static properties by using the one or more programmable objects to
10 call the one or more objects that support only static properties.



11
12 **46. (ORIGINAL)** A property value-changing method comprising:

13 programming a programmable object with property data that defines when
14 certain property value changes are to be made and what those property value
15 changes are, the property value changes being associated with rendering of a
16 multi-media editing project;

17 calling, with the programmable object, one or more objects that do not
18 support dynamic properties; and

19 responsive to said calling, using the property data to effect a property value
20 change on the one or more objects.

21
22 **47. (ORIGINAL)** The method of claim 46 further comprising calling
23 the programmable object with a current time, the programmable object using the
24 current time to ascertain when to call the one or more objects.

1 **48. (ORIGINAL)** The method of claim 46, wherein said programming
2 comprises arranging the property data in a data structure array comprising one or
3 more sets of data structures, each set of data structures being associated with a
4 property whose value is to be changed.

5
6 **49. (ORIGINAL)** The method of claim 46, wherein the property data
7 defines how the property value changes are to be made.

8
9 **50. (ORIGINAL)** One or more computer-readable media having
10 computer-readable instructions thereon which, when executed by a computer,
11 implement the method of claim 46.

12
13 **51. (ORIGINAL)** A property value-changing method comprising:
14 providing one or more objects that support only static properties; and
15 simulating dynamic properties with the one or more objects by changing
16 one or more property values at a pre-programmed time.

17
18 **52. (ORIGINAL)** The method of claim 51, wherein said simulating
19 comprises pre-programming at least one property value change, a time at which
20 the property value is to be changed, and a manner in which the property value
21 change it to take place.

22
23 **53. (ORIGINAL)** The method of claim 52, wherein said pre-
24 programming comprises pre-programming a computer-implemented object to call
25

1 the one or more objects at an appropriate time to change the one or more property
2 values.

3
4 **54. (ORIGINAL)** Software code comprising a multi-media project
5 editing application configured to implement the method of claim 51.

6
7 **55. (ORIGINAL)** A multi-media system comprising:
8 an application program configured to enable a user to define a multi-media
9 project in which multiple digital source streams can be combined;

10 a software-implemented matrix switch having multiple input pins and
11 multiple output pins, the input pins being individually associated with inputs that
12 can compete, during a common time period, for a particular output pin that is
13 associated with the matrix switch, the switch being configured to receive, at its
14 input pins, digital source streams;

15 a first data structure associated with the matrix switch and configured for
16 use in programming the matrix switch to provide a routing scheme for routing
17 input pins to output pins such that at any given time, only one input pin is routed
18 to the particular output pin;

19 a second data structure associated with and different from the first data
20 structure, the second data structure representing a user-defined multi-media project
21 and being configured so that the first data structure can be derived therefrom;

22 one or more first objects associated with the matrix switch, the one or more
23 first objects supporting only static properties associated with rendering of a multi-
24 media project; and
25

1 one or more second objects associated with the one or more first objects
2 and configured to call the one or more first objects to effect one or more property
3 value changes on the one or more first objects in a manner that makes the one or
4 more first objects appear as if they support dynamic properties.

5
6 **56. (ORIGINAL)** The multi-media system of claim 55 further
7 comprising one or more data structures associated with the programmable
8 object(s), individual data structures containing data that is to be used by the
9 programmable object(s) to effect a property value change.

10
11 **57. (ORIGINAL)** The multi-media system of claim 56, wherein the one
12 or more data structures comprise an array of one or more sets of data structures,
13 each set of data structures being associated with a property value that is to be
14 changed and containing property data that is to be used to change that property
15 value.

16
17 **58. (ORIGINAL)** The multi-media system of claim 56, wherein the one
18 or more data structures comprise an array of one or more sets of data structures,
19 each set of data structures being associated with a property whose value is to be
20 changed and containing property data that is to be used to change that property
21 value, the property data comprising: a property value that is to be changed, a time
22 at which the property value is to be changed, and a manner in which the property
23 value is to be changed.
